

ARGUMENT

Claims 1-34 are pending in the application.

Objection

The Office action objects that the word “tangible” which was added to the claims is not defined in the specification.

The applicant is confident that the word “tangible” is understood and no description is needed. “[T]he ordinary and customary meaning of a claim term is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e., as of the effective filing date of the patent application.” Phillips v. AWH Corp., 415 F.3d 1303, 1313, 75 USPQ2d 1321, 1326 (Fed. Cir. 2005) (en banc). See also MPEP 2111.01 entitled “Plain Meaning.” The applicant means tangible to have its plain meaning, i.e., something that can be touched, held, carried, moved, etc.

§101 Rejection

Claims 27-34 were rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter. Specifically, the claims were objected to as possibly covering a carrier wave. Accordingly, claims 27, 30-34 have been amended to call for a tangible computer storage medium as requested in the Office action. Accordingly, the applicant believes the section 101 rejection has been addressed and should be withdrawn.

§102(e) Rejection

Claims 1 -34 are rejected under 35 U.S.C. §102(e) as being anticipated by Shiomi et al., U.S. Publication number 2006 0095919 A1 (“Shiomi”). Shiomi describes a Java middleware unit 33 which includes a virtual machine (VM, element 33a in Fig. 15), an application managing unit (element 33b in Fig. 15) and a class library storing unit (element 33c in Fig. 15). It also has an OS unit 34 that has a kernel (element 34a) and a hardware unit that has a CPU (element 35a).

In order to anticipate a claim, all the elements of the claim must be found in a single reference. On first glance, the Shiomi reference looks similar to the pending claims. Only upon close analysis do the distinctions between the pending claims and the Shiomi reference become apparent. In general, Shiomi describes a virtual machine that is not concerned with whether a process is a kernel process or an application process like in the pending claims.

CLAIM 1

In all of the pending independent claims, a determination is made whether a process is a kernel process or an application process. The claimed element of determining whether a process is a kernel process or a user process is not found in Shiomi.

The Office action cites numerous locations for determining whether a process is a kernel process or a user process, specifically: Fig. 15; elements 33a and 33b; paragraphs 161, 162, 163; Fig. 1; paragraph 88, 89 and 95. The Figures and paragraphs that were cited to anticipate the element of determining whether a process is a kernel process or a user process do not disclose the claimed element of determining whether a process is a kernel process or a user process which makes sense as Shiomi is not concerned with whether a process is a user process or a kernel process.

Elements 33a and 33b of Fig. 15 are part of the Java Middleware Unit. None of these elements disclose making a determination between a kernel process and a user process as claimed.

Paragraph 161 discloses the library unit performing resource management. The problem is that the task ID does not make a determination of whether a process is a kernel process or a user process. Paragraph 162 discloses using a special thread to reserve necessary resources. Again, there is no disclosure of making a determination whether a thread is a user thread or a kernel thread. Paragraph 163 talks about using the special threads to reserve resources for the VM unit, but again, does not disclose making a determination of whether a process is a kernel process or a user process.

Fig. 1 illustrates inside the kernel unit an identifier providing unit 13a. The identifier provides an identification to a process but there is no determination whether a process is a kernel process or a user process. Shiomi just is not concerned with this distinction.

Paragraph 88 describes a registration callback function. The callback function reports the application id of an application when the application is complete or suspended but there is no determination whether a process is a kernel process or a user process.

Paragraph 89 describes Fig. 2 which is a program that defines the callback function from paragraph 88. As stated, the callback function has no return values and takes an integer (the application id) as an argument but there is no determination whether a process is a kernel process or a user process.

Paragraph 95 describes the library unit acquiring an application ID of the application which has requested a resource. The acquired application ID and resource name are stored in a table but there is no determination whether a process is a kernel process or a user process.

None of these cited selections disclose determining whether the process is a kernel process or a user process. The different tasks have IDs and the different applications have IDs but there is no indication or suggestion that a determination is made whether the process is a kernel process or a user process. This makes sense as Shiomi simply is not concerned with whether a process is a kernel process or a user process.

The element of making a determination as to whether a process is a user process or a kernel process is not insignificant. One of the goals of the pending application and claims is to cover allocating kernel processes which are normally thought of as overhead. By making the separation from kernel processes to user processes, the ability to track the cost of kernel processes is now possible.

Of additional note, claim 4 is very, very specific in calling for the tag value to be stored in a first word of the tag. The cited sections (Fig. 27 and paragraph 187) make no mention of storing a tag in the first word. The elements of the claim simply are not disclosed. A *prima facie* rejection has not been made.

Similarly, claim 7 is very, very specific in calling for generating a tag value to identify kernel resources allocated to a user process, saving the tag value to a first word of the tag and saving a user process identifier to identify the process to a second word of the tag. Fig. 12 and paragraph 117 are cited for disclosing these elements. Neither of these cited sections discloses the elements of the claims. The elements of the claim simply are not disclosed. A *prima facie* rejection has not been made.

As claims 2- 7 are dependent on claim 1 and an element is missing from claim 1, this element is missing from claims 2-7 and claims 2-7 also are not anticipated.

CLAIM 8

In regard to claim 8, the language is a little different, but the result is the same. In claim 8, the method calls for determining whether the process is a first defined process or a second defined process. Again, Shiomi does not make such a determination. The Office action sites to the same sections of Shiomi for disclosing this element. The Applicant refers to the previous paragraphs for a discussion of the cited sections and the failure to disclose the element of making the determination of the type of process.

As claims 9-13 are dependent on claim 8 and an element is missing from claim 8, this element is missing from claims 9-13 and claims 9-13 also are not anticipated.

CLAIM 14

Claim 14 calls for saving a tag to identify the kernel resources allocated to the user process. Similar to claim 1, Shiomi does not make a determination between pure kernel processes and user processes so Shiomi cannot determine the kernel resources allocated to the user process.

In addition, claim 15 calls for “the tag value is saved in a first word of the tag, and the user process identifier is saved in a second word of the tag.” Similar to claim 7, such an arrangement is not present in Shiomi. A *prima facie* rejection has not been made.

Similarly, claim 19 calls for “wherein said step of saving a user process identifier further comprises the steps of: extending the tag with a second word; and, saving the user process identifier to the second word of the tag.” Similar to claim 7, such an arrangement is not present in Shiomi. A *prima facie* rejection has not been made.

Claim 20 further limits the claims but calling for saving the tag value to a first word of the tag. Similar to claim 7, such an arrangement is not present in Shiomi. A *prima facie* rejection has not been made.

As claims 15-20 are dependent on claim 14 and an element is missing from claim 14, this element is missing from claims 15-20 and claims 15-20 also are not anticipated.

CLAIM 21

Claim 21 calls for identifying a type of the kernel resources allocated to the process indicated by the selected tag. Similar to claim 1, Shiomi does not identify a type of kernel resource allocated to the process. Shiomi does not separate the types of processes.

Claim 21 also calls for identifying an amount of kernel resources allocated to a process. The Office action cites to paragraph 200 and Fig. 29 as disclosing this element. These cited sections describe the heap memory management table. It does not disclose the claimed element of identifying an amount of kernel resources allocated to a process. Accordingly, a *prima facie* case of anticipation has not been made.

As claims 22-24 are dependent on claim 21 and an element is missing from claim 21, this element is missing from claims 22-24 and claims 22-24 also are not anticipated.

CLAIM 25

Claim 25 calls for a flag to indicate whether the process is a first defined process or a second defined process. Again, Shiomi does not make such a determination or use such a flag. The Office action sites to the same sections of Shiomi for disclosing this element. The Applicant refers to the previous paragraphs for a discussion of the cited sections and the failure to disclose the element of making the determination of the type of process.

Claim 25 also calls for identifying an amount of kernel resources allocated to a process. The Office action cites to paragraph 209 and Fig. 30 as disclosing this element. These cited sections describe the heap memory management area. They do not disclose the claimed element of identifying an amount of kernel resources allocated to a process. Accordingly, a *prima facie* case of anticipation has not been made.

Claim 25 also calls for “determining whether the amount of kernel resource usage exceeds the threshold limits according to the user policy; and, taking an action according to the usage policy when the amount of kernel resource usage is over the threshold limits.” The Office action cites to paragraph 209 and Fig. 30 as disclosing this element. These cited sections describe the heap memory management area and do not describe a threshold or taking steps if a threshold is exceeded. Accordingly, a *prima facie* case of anticipation has not been made.

As claim 26 is dependent on claim 25 and an element is missing from claim 25, this element is missing from claim 26 and claim 26 also is not anticipated.

CLAIM 27

Claim 27 calls for a data structure with a first field identifying user processes and a second field identifying kernel resources allocated to the user processes. Similar to claim 1, as Shiomi does not differentiate between user processes and kernel processes as called for in the claims, this element is missing.

Claim 27 calls for a data structure a first field containing a user process identifier to identify a user process allocated with kernel resources and a second field containing a value to identify kernel resources allocated to the user process. The Office action cites to paragraph 187 and Fig. 27 for disclosing this element. Paragraph 187 simple states data is stored. Fig. 27 illustrates data being stored. There is no disclosure of a data structure with a first field containing a user process identifier to identify a user process allocated with kernel resources and a second field containing a value to identify kernel resources allocated to the user process. Accordingly, a *prima facie* case of anticipation has not been made.

Claim 29 calls for the first field is a first long word of a tag, and the second field is a second long word of the tag. The Office action cites to paragraph 117 and Fig. 12 for disclosing this element. Paragraph 117 and Fig. 12 describe a class. There is no disclosure of a first word or a second word or what is stored in each word. Accordingly, a *prima facie* case of anticipation has not been made.

As claims 28-29 are dependent on claim 27 and an element is missing from claim 27, this element is missing from claims 28-29 and claims 28-29 also are not anticipated.

CLAIM 30

Claim 30 calls for determining whether the process is a kernel process or a user process. Similar to claim 1, as Shiomi does not differentiate between user processes and kernel processes as called for in the claims, this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated.

CLAIM 31

Claim 31 calls for determining whether the process is a first defined process or a second defined process. Similar to claim 1, as Shiomi does not differentiate between a first process and a second process as called for in the claims, this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated.

CLAIM 32

Claim 32 calls for tagging user processes that are allocated with kernel resources. Similar to claim 1, as Shiomi does not differentiate between user processes and kernel processes as called for in the claims, this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated.

CLAIM 33

Claim 33 calls for identifying an amount of kernel resources allocated to a process indicated by a selected tag. Similar to claim 1, as Shiomi does not differentiate between user processes and kernel processes as called for in the claims, this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated.

Claim 33 also calls for identifying an amount of kernel resources allocated to a process. The Office action cites to paragraph 215 and Fig. 32 as disclosing this element. These cited sections describe the heap memory management area. They do not disclose the claimed element of identifying an amount of kernel resources allocated to a process. Accordingly, a *prima facie* case of anticipation has not been made.

CLAIM 34

Claim 34 calls for a flag indicating whether the process is a first defined process or a second defined process. Similar to claim 1, as Shiomi does not differentiate between a first process and a second process as called for in the claims, this element is missing. As an element in the claims is not in Shiomi, the claim is not anticipated.

Claim 34 also calls for identifying an amount of kernel resources allocated to a process. The Office action cites to paragraph 209 and Fig. 30 as disclosing this element. These cited sections describe the heap memory management area. They do not disclose the claimed element of identifying an amount of kernel resources allocated to a process. Accordingly, a *prima facie* case of anticipation has not been made.

Claim 34 also calls for “determining whether the amount of kernel resource usage exceeds the threshold limits according to the user policy; and, taking an action according to the usage policy when the amount of kernel resource usage is over the threshold limits.” The Office action cites to paragraph 209 and Fig. 30 as disclosing this element. These cited sections describe the heap memory management area and do not describe a threshold or taking steps if a threshold is exceeded. Accordingly, a *prima facie* case of anticipation has not been made.

REMARKS

In view of the above amendment, applicant believes the pending application is in condition for allowance. If the Examiner has any questions or suggestions, the Examiner is encouraged to call the applicant direct at 312-474-6610.

Dated: May 21, 2007

Respectfully submitted,

By /W. J. Kramer/
William J. Kramer

Registration No.: 46,229
MARSHALL, GERSTEIN & BORUN
233 S. Wacker Drive, Suite 6300
Sears Tower
Chicago, Illinois 60606-6357
(312) 474-6300
Attorney for Applicant